

CLAIMS

WHAT IS CLAIMED IS:

1. A shared computer network storage system, comprising:
 - a first database containing file data;
 - a second database containing information (metadata) about said file data of said first database;
 - a server, said server executing file commands on said first file database, said server contemporaneously updating said second metadatabase upon executing said file commands; and
 - a client application, said client application communicating with said server, said client application invoking file commands upon said server, said server executing said file commands and updating information regarding said first file and second metadata databases displayed by said client application; whereby
said client application controls files in said first file database and information regarding status of said first database files is more readily available by reference to said second metadatabase.
2. The shared computer network storage system of claim 1, wherein said first file database is distributed over at least two physical storage devices.
3. The shared computer network storage system of claim 1, wherein said second

2 metadatabase is distributed over at least two physical storage devices.

4. The shared computer network storage system of claim 1, wherein said client
2 application communicates with said server via a proxy.

5. The shared computer network storage system of claim 1, wherein said server
2 comprises a non-routable network.

6. The shared computer network storage system of claim 1, wherein said server
2 comprises a transaction processor.

7. The shared computer network storage system of claim 6, wherein said
2 transaction processor guarantees access to and transactions on said first and second
databases.

8. The shared computer network storage system of claim 1, wherein said server
2 comprises an enterprise java bean cluster (EJBC).

9. The shared computer network storage system of claim 8, wherein said enterprise
2 java bean cluster (EJBC) handles business logic and resource access methods as well as
memory caching for common resources.

10. The shared computer network storage system of claim 1, wherein said server

2 further comprises an application network.

11. The shared computer network storage system of claim 10, wherein said
2 application network further comprises a java application cluster.

12. The shared computer network storage system of claim 10, wherein said
2 application network handles display functions and resource requests.

13. The shared computer network storage system of claim 1, wherein said server
further comprises a web server.

14. The shared computer network storage system of claim 13, wherein said web
2 server handles all requests for static content and proxies requests for dynamic content.

15. The shared computer network storage system of claim 1, wherein said server
further comprises a load balancer, said load balancer proxying requests to a sub-server
having the highest degree of availability or functionality.

16. The shared computer network storage system of claim 1 wherein said server
2 further comprises a DNS redirector, said DNS redirector proxying requests to a
resource having a highest degree of functionality.

17. The shared computer network storage system of claim 1 wherein said server

2 further comprises:

4 a transaction processor, said transaction processor on a non-routable
network, said transaction processor guarantees access to and transactions on said
first and second databases;

6 an enterprise java bean cluster (EJBC) on a non-routable network, said
enterprise java bean cluster (EJBC) coupled to said transaction processor and
8 handling business logic and resource access methods as well as memory caching
for common resources;

10 an application network on a non-routable network, said application
network coupled to said enterprise java bean cluster, said application network
including a java application cluster and handling display functions and resource
requests;

12 a web server, said web server coupled to said application network and
handling all requests for static content and proxies requests for dynamic content;

14 a load balancer, said load balancer coupled to said web server and
proxying requests to a sub-server having the highest degree of availability or
16 functionality; and

18 a DNS redirector, said DNS redirector coupled to said load balancer and
proxying requests to a resource having a highest degree of functionality.
20

18. The shared computer network storage system of claim 1, wherein said client
2 application is web-based.

19. The shared computer network storage system of claim 1, wherein said client
application interacts with an operating system running upon a computer upon which
said client application is also running, said client application adopting and implementing
a visual display format similar to said operating system.

20. A shared computer network storage system, comprising:

a first database containing file data, said first database distributed over at
least two physical storage devices;

a second database containing information (metadata) about said file data
of said first database, said second database distributed over at least two physical
storage devices;

a server, said server executing file commands on said first file database,
said server contemporaneously updating said second metadatabase upon
executing said file commands, said server including:

a transaction processor, said transaction processor on a non-routable
network, said transaction processor guarantees access to and transactions on said
first and second databases;

an enterprise java bean cluster (EJBC) on a non-routable network, said
enterprise java bean cluster (EJBC) coupled to said transaction processor and
handling business logic and resource access methods as well as memory caching
for common resources;

an application network on a non-routable network, said application
network coupled to said enterprise java bean cluster, said application network

including a java application cluster and handling display functions and resource requests;

a web server, said web server coupled to said application network and handling all requests for static content and proxies requests for dynamic content;

a load balancer, said load balancer coupled to said web server and proxying requests to a sub-server having the highest degree of availability or functionality; and

a DNS redirector, said DNS redirector coupled to said load balancer and proxying requests to a resource having a highest degree of functionality; and

a client application, said client application communicating with said server via a proxy, said client application invoking file commands upon said server, said server executing said file commands and updating information regarding said first file and second metadata databases displayed by said client application; whereby

said client application controls files in said first file database and information regarding status of said first database files is more readily available by reference to said second metadatabase.

21. The shared computer network storage system of claim 20, wherein said client application is web-based.

22. The shared computer network storage system of claim 20, wherein said client application interacts with an operating system running upon a computer upon which

said client application is also running, said client application adopting and implementing a visual display format similar to said operating system.

23. A method for providing private file space and information transfer over a public computer network, the steps comprising:

providing a publicly-available private file space system coupled to the public computer network;

providing a client program in communication with the public computer network;

sending a request from said client program to said publicly-available private file space system ("private system");

evaluating said request;

authenticating said request;

satisfying said request; and

returning a success indicator to said client program indicating the success or failure of said request; whereby

said client program may create and control files held by said private system.

24. The method for providing private file space and information transfer over a public computer network as set forth in claim 23, wherein the step of evaluating said request further comprises evaluating said request for static content and returning an appropriate response if said request is for static content.

25. The method for providing private file space and information transfer over a public computer network as set forth in claim 24, the steps further comprising:

providing an application network within said private system;
proxying said request to said application network; and
parsing a header of said request.

26. The method for providing private file space and information transfer over a public computer network as set forth in claim 23, wherein said step of authenticating said request further comprises:

authenticating a user using said client program; and
authenticating said request made by said client program to ensure that it conforms with an account associated with said user.

27. The method for providing private file space and information transfer over a public computer network as set forth in claim 23, further comprising:

parsing multipart form data associated with said request;
determining said request's type; and
submitting said request.

28. A method for providing private file space and information transfer over a public computer network, the steps comprising:

providing a publicly-available private file space system coupled to the public computer network;

providing a client program in communication with the public computer
network;

sending a request from said client program to said publicly-available
private file space system ("private system");

evaluating said request for static content and returning an appropriate
response if said request is for static content;

providing an application network within said private system;

proxying said request to said application network; and

parsing a header of said request

authenticating said request by authenticating a user using said client
program and authenticating said request made by said client program to ensure
that it conforms with an account associated with said user;

parsing multipart form data associated with said request;

determining said request's type;

submitting said request;

satisfying said request; and

returning a success indicator to said client program indicating the success
or failure of said request; whereby

said client program may create and control files held by said private
system.

29. A data structure for effecting file operations on a private file space and
information transfer system over a public computer network, comprising:

a user data object;

a process request object; and

a recovery object;

said user information object, said process request object, and said recovery object associated within a file action object.

30. The data structure for effecting file operations on a private file space and information transfer system over a public computer network as set forth in claim 29, wherein said user data object further comprises:

a user information object; and

a security object.

31. The data structure for effecting file operations on a private file space and information transfer system over a public computer network as set forth in claim 29, wherein said process request object further comprises:

a file operation object comprising said recovery object and a database IO object, a file IO object, and an administration object.

32. The data structure for effecting file operations on a private file space and information transfer system over a public computer network as set forth in claim 29, wherein said recovery object further comprises:

a recovery IO object;

a mount status object;

6 a recovery administration object; and
a recovery process object.

33. A data structure for effecting file operations on a private file space and
2 information transfer system over a public computer network, comprising:

a user data object, said user data object having a user information object;
4 and a security object;

a process request object, said process request object including a file
operation object, a database IO object, a file IO object, and an administration
6 object; and

a recovery object, said recovery object incorporated in said file operation
object, said recovery object including a recovery IO object, a mount status
8 object, a recovery administration object, and a recovery process object;

said user information object, said process request object, and said
2 recovery object associated within a file action object; whereby

file operations may be facilitated by the data structure including recovery
14 from resource failure.

15 34. A shared file storage resource for a computer network, comprising:

2 an allocatable file storage resource;

a server, said server coupled to said storage resource, said server:

4 allocating individual user file space for a plurality of users on said
storage resource;

6 receiving files for storage on said storage resource;
transmitting files stored on said storage resource;
8 generating control-protocol codes for transmitting said files;
receiving file commands for controlling files on said storage resource;
10 and
transmitting display codes indicating file status on said storage resource,
12 said display codes representing said storage resource as a network drive;
a first network connection, said first network connection coupling said
14 server to the computer network;
a workstation, said workstation:
6 receiving files for storage on said storage resource;
transmitting files stored on said storage resource;
18 receiving file commands for controlling files on said storage resource;
and
20 transmitting display codes indicating file status on said storage resource,
said display codes representing said storage resource as a network drive;
22 whereby
a user may store, retrieve, and control files in a unique and secure file
24 storage area on said allocatable storage resource available throughout the
computer network and detached from said workstation.

35. The shared file storage resource for a computer network as set forth in claim 34,

2 wherein said display codes further comprise:

a browser-interpretable object, such as a JavaScript object, said object
displaying file status on said storage resource as a web page.

36. The shared file storage resource for a computer network as set forth in claim 34,
further comprising:

a standalone program running on said workstation, said standalone
program interpreting said display codes and providing a seamless interface to
said user, said seamless interface presenting said storage resource as a local or
network resource to said user and allowing said user to manipulate files on said
storage resource in the same manner as local storage resources such as a floppy
disk drive or a local hard drive.

37. The shared file storage resource for a computer network as set forth in claim 34,
wherein said computer network, further comprises:

the Internet.

38. A method for transferring data from a first network resource to a second
network resource at the direction of a user, the steps comprising:

submitting a first file location indicating data to be transferred to the
second network resource;

the second network resource requesting said data at said first file location
from the first network resource;

the first network resource transmitting said data to the second network

8 resource; and

the second network resource notifying the user of successful transfer
10 upon successful reception of said data; whereby

the user may use the first and second network resources to obtain and
12 control said data.

39. The method for transferring data as set forth in claim 38, wherein the second
2 network resource comprises a subscriber-based system of network-available storage
space.

40. The method for transferring data as set forth in claim 38, wherein the first and
second network resources are coupled to the Internet.

41. The method for transferring data as set forth in claim 38, the steps further
comprising:

displaying to the user a status of transmission of said data from said first
4 network resource to said second network resource.

42. The method for transferring data as set forth in claim 38, the steps further
2 comprising:

verifying the user as a subscriber to or member of the second network
4 resource.

43. A method for transferring data from a first network resource to a second
network resource at the direction of a user, the steps comprising:

submitting a first file location indicating data to be transferred to the
second network resource, the second network resource being a subscriber-based
system of network-available data storage space;

verifying the user as a subscriber to or member of the second network
resource;

the second network resource requesting said data at said first file location
from the first network resource;

the first network resource transmitting said data to the second network
resource via Internet;

displaying to the user a status of transmission of said data from said first
network resource to said second network resource; and

the second network resource notifying the user of successful transfer
upon successful reception of said data; whereby

the user may use the first and second network resources to obtain and
control said data.

44. A client-server system for a network-based data storage and manipulation
system, comprising:

a client system, said client system having a file access service and a file
manipulation service;

a server, said server providing network-based data storage resources and

6 responding to requests transmitted by said client system, said server effecting
said requests;

8 said server determining if a client request is one for metadata regarding
data stored upon said server;

10 said server providing said metadata if said client request is for metadata
and transmitting said metadata to said file manipulation service; and

12 said server performing a file action if said client request is not for
metadata, said server updating said metadata and transmitting said metadata to
said file manipulation service; whereby

14 said server operates, and said client system presents, operations on said
server in a manner similar to operations local to said client system.

45. The client-server system for a network-based data storage and manipulation
system as set forth in claim 44, wherein said file access service further comprises:

a request processing layer for processing requests; and

a first network I/O layer for transmitting said requests to said server.

46. The client-server system for a network-based data storage and manipulation
system as set forth in claim 44, wherein said file manipulation service further
comprises:

a parser, said parser parsing said metadata from said server;

a data structure, said data structure receiving and preserving parsed data
from said parser; and

a data display layer, said data display layer operating upon and
displaying said parsed data; whereby

metadata may be displayed to inform about data stored upon said server.

47. The client-server system for a network-based data storage and manipulation
system as set forth in claim 46, wherein said parser is an XML parser.

48. The client-server system for a network-based data storage and manipulation
system as set forth in claim 44, wherein said server further comprises:

a second network I/O layer, said second network I/O layer engaged
when said requests are not for metadata, said second network I/O layer
transmitting requests for file action; and

a resource access layer, said resource access layer receiving
transmissions from said second network I/O layer and effecting said requests,
said resource access layer engaged when said requests are for metadata, said
resource access layer obtaining and transmitting said metadata; and

a metadata compiler, said metadata compiler receiving said metadata
from said resource access layer, compiling said metadata, and transmitting said
compiled metadata to said client system.

49. The client-server system for a network-based data storage and manipulation
system as set forth in claim 48, wherein said metadata compiler is an XML generator.

50. A client-server system for a network-based data storage and manipulation
system, comprising:

a client system, said client system having a file access service and a file
manipulation service;

a server, said server providing network-based data storage resources,
said server creating and maintaining metadata regarding stored data, said server
responding to requests transmitted by said client system, said server effecting
said requests;

said server determining if a client request is one for metadata;

said server providing said metadata if said client request is for metadata
and transmitting said metadata to said file manipulation service;

said server performing a file action if said client request is not for
metadata, said server updating said metadata and transmitting said metadata to
said file manipulation service;

said file access service having a request processing layer for processing
requests and a first network I/O layer for transmitting said requests to said
server;

said file manipulation service having an XML parser, said XML parser
parsing said metadata from said server, said file manipulation service having a
data structure, said data structure receiving and preserving parsed data from said
parser, and said file manipulation service having a data display layer, said data
display layer operating upon and displaying said parsed data so that metadata
may be displayed to inform about data stored upon said server; and

24 said server having a second network I/O layer, said second network I/O
layer engaged when said requests are not for metadata, said second network I/O
26 layer transmitting requests for file action, said server having a resource access
layer, said resource access layer receiving transmissions from said second
28 network I/O layer and effecting said requests, said resource access layer
engaged when said requests are for metadata, said resource access layer
30 obtaining and transmitting said metadata, and said server having a metadata
compiler in the form of an XML generator, said metadata compiler receiving
32 said metadata from said resource access layer, compiling said metadata, and
transmitting said compiled metadata to said client system; whereby

 said server operates as and said client system presents operations on said
server in a manner similar to operations local to said client system.